**Введение**

**1. Основные понятия:**

* Что такое контейнеры и как они отличаются от виртуальных машин.
* Понятия образов, контейнеров, Dockerfile, Docker Hub.
* Установка Docker на различных платформах (Windows, macOS, Linux).

**Практика:**

* Установите Docker на вашу рабочую машину.

**Основы Docker**

**2. Основные команды Docker:**

* **docker run**: запуск контейнеров.
* **docker ps**: просмотр запущенных контейнеров.
* **docker stop** и **docker rm**: остановка и удаление контейнеров.
* **docker images**: просмотр образов.
* **docker rmi**: удаление образов.

**Практика:**

* Запустите первый контейнер с использованием официального образа (например, **hello-world**).

**Работа с образами**

**3. Управление образами:**

* **docker pull**: загрузка образов.
* **docker build**: создание собственных образов из Dockerfile.
* Понятие слоев в Docker.

**Практика:**

* Загрузите и запустите несколько различных образов.
* Создайте собственный образ с использованием Dockerfile.

**Работа с контейнерами**

**4. Управление контейнерами:**

* Понимание изоляции и управление ресурсами (CPU, память).
* Объём данных и монтирование томов.
* Сетевые настройки контейнеров.

**Практика:**

* Запустите несколько контейнеров с разными настройками ресурсов и сетевыми конфигурациями.
* Используйте тома для сохранения данных контейнеров.

**Docker Compose**

**5. Docker Compose:**

* Понятие Docker Compose.
* Написание **docker-compose.yml** файлов.
* Команды **docker-compose up**, **docker-compose down**.

**Практика:**

* Напишите **docker-compose.yml** для запуска простого приложения, состоящего из нескольких контейнеров (например, веб-сервер и база данных).

**Оркестрация и масштабирование**

**6. Docker Swarm и Kubernetes:**

* Введение в Docker Swarm: настройка кластера, управление сервисами.
* Введение в Kubernetes: основные понятия (Pods, Deployments, Services), установка мини-кластера (minikube).

**Практика:**

* Настройте простой кластер с Docker Swarm и разверните на нем приложение.
* Настройте minikube и разверните простое приложение на Kubernetes.

**Безопасность**

**7. Безопасность Docker:**

* Практики безопасного создания образов.
* Управление секретами.
* Контроль доступа.

**Практика:**

* Создайте Docker образы с соблюдением практик безопасности.
* Используйте секреты в Docker Compose или Kubernetes для защиты чувствительных данных.

**Интеграция и DevOps**

**8. CI/CD и интеграция с другими инструментами:**

* Настройка CI/CD пайплайнов с использованием Docker (например, Jenkins, GitLab CI).
* Интеграция Docker с инструментами мониторинга и логирования (например, Prometheus, ELK stack).

**Практика:**

* Настройте CI/CD пайплайн для вашего проекта с использованием Docker.
* Интегрируйте мониторинг и логирование для вашего контейнеризированного приложения.

**Реальные проекты и кейсы**

**9. Реальные проекты:**

* Применение всех изученных навыков в реальных проектах.
* Создание и развертывание сложных многокомпонентных приложений.

**Практика:**

* Разработайте и разверните комплексное приложение, используя все аспекты Docker и Docker Compose, возможно с оркестрацией через Kubernetes.

**Дополнительные ресурсы и материалы**

**10. Ресурсы:**

* Официальная документация Docker: https://docs.docker.com/
* Курсы и видеоуроки на платформах Udemy, Coursera, Pluralsight.
* Книги: "Docker Deep Dive" by Nigel Poulton, "Docker: Up & Running" by Karl Matthias and Sean P. Kane.

Этот план позволит вам систематически изучить Docker, начиная с основ и постепенно переходя к более сложным темам, включая оркестрацию и безопасность.